(FILE 'USPAT' ENTERED AT 14:16:47 ON 09 OCT 1997)
SET PAGE SCROLL
L1 58 S 395/286/CCLS
L2 9 S L1 AND UART
L3 1 S L2 AND FULL DUPLEX
L4 1 S L3 AND FIFO

=> d

1. 5,564,061, Oct. 8, 1996, Reconfigurable architecture for multi-protocol data communications having selection means and a plurality of register sets; Eric Davies, et al., 395/884, **286**, 310, 311, 500, 831 [IMAGE AVAILABLE]

=> d kwic

US PAT NO: 5,564,061 [IMAGE AVAILABLE] L4: 1 of 1
US-CL-CURRENT: 395/884, **286**, 310, 311, 500, 831

SUMMARY:

BSUM(14)

The present invention may be configured to provide a 550 type **UART** in parallel with an 8530 type device for synchronous/asynchronous communication or configured to provide a 550 device in series with. . .

SUMMARY:

BSUM(15)

This . . . designed to allow a CPU and a local protocol controller to interface, while appearing to be communicating with a standard **UART** interface. The controller can then use a serial communications controller (SCC) block for any synchronous or asynchronous protocol. In an. . . embodiment, the present invention may be configured to allow a single device to be programmable to behave either as asynchronous **UART** or synchronous USART.

DETDESC:

DETD(3)

FIG. . . . line 10, includes in part a CPU bus 12. The CPU bus 12 provides a parallel output 13 to a **UART** 14. The **UART** converts the parallel stream into a serial stream 15 for output through a serial port. The serial stream 15 is. . .

DETDESC:

DETD(4)

The **UART** 14 may be implemented with a device such as the SSi 73M550 **UART**, manufactured by Silicon Systems, Inc., of Tustin, Calif., assignee of

the present patent application. This device, and others like it, shall be referred to herein as "550 type devices" or "550 type **UART**'s".

DETDESC:

DETD(5)

Often, . . . parallel format. Therefore, present day communication devices include a second converting means for converting the serial output of a 550 **UART** to parallel data for use by the processor. The processor provides parallel output which then must be reconverted to serial. . .

DETDESC:

DETD(6)

In . . . this application, such devices are referred to as 8530-type SCC devices or 8530-type controllers. The 8530 SCC typically includes two **full duplex** channe
ls. That is, each channel can receive serial data and convert it to parallel or receive parallel data and convert. . .

DETDESC:

DETD(7)

Block 51 represents an asynchronous communications device such as the SSi 73M550 **UART** or the SSi 73M450 **UART** manufactured by Silicon Systems, Inc. of Tustin, Calif., assignee of the present patent application. This device, and others like it, shall be referred to herein as "450 type devices" or "450 type **UART**'s". A device implementing both a 550 type **UART** and a 450 type **UART** is said to have a "450/550 interface." The synchronous communications device 50 provides output 52 to multiplexer 54. The asynchronous. . .

DETDESC:

DETD(8)

It is often desired to provide a system where a 550 type **UART** is coupled in series with an 8530 type **UART**. This is often the case when a communications device has an associated processor. An example of a prior art scheme implementing a 550 **UART** and an 8530 SCC in connection with a microprocessor is illustrated in FIG. 4. A CPU bus 12 is coupled to a 550 **UART**. The 550 **UART** consists of two components, a 550 register set 16 and a 550 serializer 17. The 550 register set 16 receives. . . 12 and provides it to the serializer 17 for conversion to serial output. The serial output 15 of the 550 **UART** is provided to one channel of an 8530 SCC such as channel A block 18.

DETDESC:

DETD(9)

The . . . registers, line status registers, transmitter holding register, modem control register, modem status register, interrupt enable register, interrupt ID register and **FIFO** control register. After data has been provided to these registers, it is provided to the serializer for conversion to a. . .

DETDESC:

DETD(11)

Many . . . communicate or transmit data on a modem are written to communicate with a register set associated with a 450 type **UART** or 8530 type device. Therefore, this invention emulates the register set of 450 and

8530 type **UART**'s, but does not require all of the circuitry associated with a prior art 450 or 550 type **UART**. The invention also provides the functionality of an 8530 SCC without requiring all of the circuitry associated with a prior. . .

DETDESC:

DETD(22)

The 550 register block 76 utilizes a number of registers to emulate the register set of a 550 type **UART**. In this invention, dual port registers are utilized so that data written into a register may be read onto a. . .

DETDESC:

DETD(27)

The transmit register 30 is implemented with a first in-first out (**FIFO**) register. The register 30 receives transmit data information from bus 26 and therefore emulates a 550 type **UART** transfer register. This data is immediately available to be read onto bus 33, and therefore register 30 emulates a channel. . .

DETDESC:

DETD(28)

Similarly, register 31 emulates a receive **FIFO** for a 550 type **UART** and a transmit **FIFO** for channel B of an 8530 device.
The interrupt logic register
32 provides interrupts to the host computer and to. . .

DETDESC:

DETD(29)

The control registers 28 of this invention emulate a number of registers in a typical 550 type **UART**, such as the SSi 73M550. The registers 28 emulate line control registers, deviser latch registers, line status registers, modem control registers, modem status registers, and **FIFO** control registers of a 550 **UART**. However, the implementation
of this invention allows a number of
elements of a prior art 550 type **UART** to be eliminated. In particular, receiver shift registers, receiver timing and control, baud rate generators, transmitter timing control, transmitter shift. . .

DETDESC:

DETD(43)

FIGS. . . . values for each register. This mapping provides a standard interface to application programs written to communicate with a 550 type **UART**. For example, prior art applications programs are written to write to eight address locations in a specific memory location. This. . .